HAL open science

# Adaptation de ressources en langue anglaise pour interroger des données tabulaires en français

Alexis Blandin

## To cite this version:

Alexis Blandin. Adaptation de ressources en langue anglaise pour interroger des données tabulaires en français. Traitement Automatique des Langues Naturelles, 2021, Lille, France. pp.1-8. hal-03265910

**HAL Id: hal-03265910**

**https://hal.science/hal-03265910v1**

Submitted on 23 Jun 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

*Actes de la 28e Conférence sur le Traitement Automatique des Langues Naturelles*
*Lille, France, 28 juin au 2 juillet 2021*

# Adaptation de ressources en langue anglaise pour interroger des données tabulaires en français

Alexis Blandin[1, 2] (1) IRISA, EXPRESSION, Vannes, France
(2) UNEEK, 44000 Nantes, France
alexis.blandin@univ-ubs.fr

RÉSUMÉ

Les récents développements des approches d'apprentissage neuronal profond ont permis des avancées très significatives dans le domaine de l'interrogation des systèmes d'information en langage naturel. Cependant, pour le français, les ressources à disposition ne permettent de considérer que les requêtes sur des données stockées sous forme de texte. Or, aujourd'hui la majorité des données utilisées en entreprise sont stockées sous forme tabulaire. Il est donc intéressant d'évaluer si les ressources anglophones associées (jeux de données tabulaires et modèles) peuvent être adaptées au français tout en conservant de bons résultats.

ABSTRACT

**Adaptation of resources in English to query French tabular data**

Research in artificial intelligence has led to significant progress in the processing of questions in French natural language through machine learning. However, for the French language, the resources available only allow us to consider queries on data stored in text form. But, available data used in companies repositories are stored in tabular form, it is therefore of particular interest to evaluate whether the available English-language resources (tabular data and models) can be effectively adapted to the French language while maintaining a good information retrieval performance.

MOTS-CLÉS : Traitement du langage naturel, recherche d'information, intelligence artificielle.

KEYWORDS: Natural language processing, information retrieval, artificial intelligence.

## 1 Introduction

Ces dernières années les avancées en traitement automatique du langage permettent d'envisager des applications plus poussées dans des milieux professionnels, comme par exemple l'amélioration du traitement des données pour la gestion de la relation client (CRM). Ainsi le traitement des questions en langage naturel sur des données peut être très prometteur.

Le traitement des questions en langage naturel a connu un vif intérêt, à l'image du jeu de données SQuAD présenté par (Rajpurkar *et al.*, 2018), qui fait désormais partie du jeu de tâche du benchmark GLUE (Wang *et al.*, 2019). La tâche consiste à extraire les portions de textes qui permettent de répondre à la question posée. Or, la plupart des données sont stockées sous forme d'une base de données tabulaire. Ainsi, dans le cadre de données stockées dans une base SQL, le contexte des questions serait une table de données, et la réponse serait alors une traduction de cette question en SQL.


Cette œuvre est mise à disposition sous licence Attribution 4.0 International.

Plusieurs jeux de données ont été réalisés afin de répondre à cette tâche d'analyse de données tabulaires, que ce soit en se focalisant sur des questions générales et complexes (Pasupat & Liang, 2015), plusieurs questions simples (Iyyer *et al.*, 2017), ou en cherchant une traduction la plus fidèle qui soit d'un langage naturel vers le SQL (Zhong *et al.*, 2017).
Si chacun de ces jeux de données propose un modèle associé, on peut toutefois remarquer que le modèle TAPAS proposé par (Herzig *et al.*, 2020), utilise d'une manière originale l'architecture BERT (Devlin *et al.*, 2019), pour proposer un nouveau traitement de cette tâche ; il donne actuellement les meilleurs résultats.
Cependant toutes ces ressources ne sont conçues que pour un usage qui concerne la langue anglaise. Par suite, l'on peut se demander dans quelle mesure une nouvelle collecte de données est nécessaire pour obtenir un modèle équivalent à TAPAS pour le français. C'est pourquoi nous proposons ici une traduction du jeu de données proposé par (Zhong *et al.*, 2017), ainsi que son évaluation sur une version ré-entraînée du modèle TAPAS, afin de déterminer si ce jeu de données obtenu par traduction d'une ressource anglophone est suffisant pour obtenir un modèle de traitement de questions en langage naturel sur des données tabulaires en français. Plus précisément, nous souhaitons établir dans quelle mesure ce modèle ré-appris sur les données traduites produit des résultats comparables à ceux obtenus pour l'anglais sur une tâche équivalente.

# 2 Présentation des jeux de données

Afin de réaliser la tâche de traduction du langage naturel en une expression SQL, plusieurs jeux de données peuvent être exploités :

— **WIKITableQuestion** (Pasupat & Liang, 2015) est un jeu de données composé de questions sur des tables HTML issues de Wikipedia auxquelles sont associées des questions complexes réalisées par des humains à qui il a été demandé de créer, suivant une table donnée, des questions complexes dont la réponse nécessite plusieurs opérations sur la table (agrégation, comparaisons, superlatifs, opérations mathématiques). Au total il comprend 22 033 questions sur 2 108 tables.

— **SQA** (Iyyer *et al.*, 2017) : cet ensemble de données a été construit en demandant à des humains de décomposer un sous-ensemble de questions hautement compositionnelles de WIKITQ, où chaque question décomposée résultante peut être renseignée par une ou plusieurs cellules d'une table SQL. L'ensemble final se compose de 6 066 séquences de questions avec 2,9 questions par séquence en moyenne.

— **WikiSQL** (Zhong *et al.*, 2017) : ce jeu de données se concentre sur la traduction de texte en SQL. Il a été construit en demandant à des humains de paraphraser une question basée sur un modèle en langage naturel, deux autres étant invités à vérifier la qualité des paraphrases proposées. Le résultat est un ensemble de 80 654 questions sur 24 241 tables issues de Wikipédia.

Entre ces trois jeux de données, notre choix s'est porté sur WikiSQL, car c'est le plus important d'un point de vue quantitatif, mais aussi parce qu'il fait office de benchmark pour cette tâche, étant souvent cité en ce sens ((Baik *et al.*, 2019), (Lyu *et al.*, 2020)). De plus dans leur article présentant le modèle, (Herzig *et al.*, 2020) ont pu tester l'apprentissage par transfert de WIKISQL vers un autre jeu de données avec un certain succès. Notre expérience étant fondée sur cette application de l'apprentissage par transfert sur des données traduites, le choix de ce jeu de donné semble justifié.

De la même manière que (Kabbadj, 2018) ont proposé une traduction du jeu de données SQuAD en

utilisant l'API de google traduction, nous proposons une version traduite du jeu de données WikiSQL en utilisant cette même API. Cette tâche de traduction comporte trois étapes :

— la traduction de la question en langage naturel de l'anglais vers le français
— la traduction des entêtes des colonnes de la table lorsque cela est nécessaire
— le remplacement des entêtes dans les requêtes SQL.

Le résultat de cette étape de traduction est illustré par les exemples présentés dans les tableaux 1 et 2.

| | Original |
|---|---|
| Question | What is the UNGEGN, when the Value is 10 000 ? |
| Headers | ['Value', 'Khmer', 'Word Form', 'UNGEGN', 'ALA-LC', 'Notes'] |
| SQL | 'SELECT UNGEGN FROM table WHERE Value = 10 000' |

TABLE 1 – Exemple d'une question du jeu de données WikiSQL, ainsi que les entêtes de la table associée, et la requête SQL correspondante

| | Traduction |
|---|---|
| Question | Qu'est-ce que l'UNGEGN, lorsque la valeur est de 10 000 ? |
| Entêtes | ['Valeur', 'Khmer', 'Forme lexicale', 'UNGEGN', 'ALA-LC', 'Notes'] |
| SQL | 'SELECT UNGEGN FROM table WHERE Valeur = 10 000' |

TABLE 2 – Exemple du résultat d'une traduction d'un item du jeu de données WikiSQL

Par ailleurs, nous avons respecté la partition du jeu de données original, à savoir, 56355 (70%) questions dans le jeu d'apprentissage, 8421 (10%) dans le jeu de validation et 1578 (20%) dans le jeu de test.

De plus, on peut appréhender la complexité des requêtes du jeu de données de deux manières : d'une part en observant la longueur des requêtes en langage naturel comme présenté dans le diagramme en figure 1, et d'autre part en observant les proportions des différents agrégateurs dans le jeu de données comme dans le diagramme en figure 2.

On remarque alors que les requêtes en langage naturel sont assez courtes (autour d'une dizaine de mots). Cette distribution est semblable à celle du jeu de données anglophone présenté par (Zhong *et al.*, 2017). De plus, on peut voir dans le diagramme en figure 2, que la majorité des requêtes SQL n'utilisent pas de fonctions d'agrégat, et ne s'assimilent donc qu'à une simple sélection sur la table. Ainsi ce jeu de données se caractérise par des requêtes d'une relative simplicité.

## 2.1 La traduction des données

Notre choix s'est porté sur les systèmes de Google Translate afin de suivre le même protocole de traduction que celui proposé par (Kabbadj, 2018) pour passer des données de Squad à SquadFr. De plus une étude récente réalisée par (Aiken, 2019) montre que l'outil est suffisamment performant pour notre cas d'usage. Toutefois, il serait intéressant d'étudier plus spécifiquement l'impact de la traduction sur les performances du modèle et ceci pourra être envisagé dans des travaux futurs.

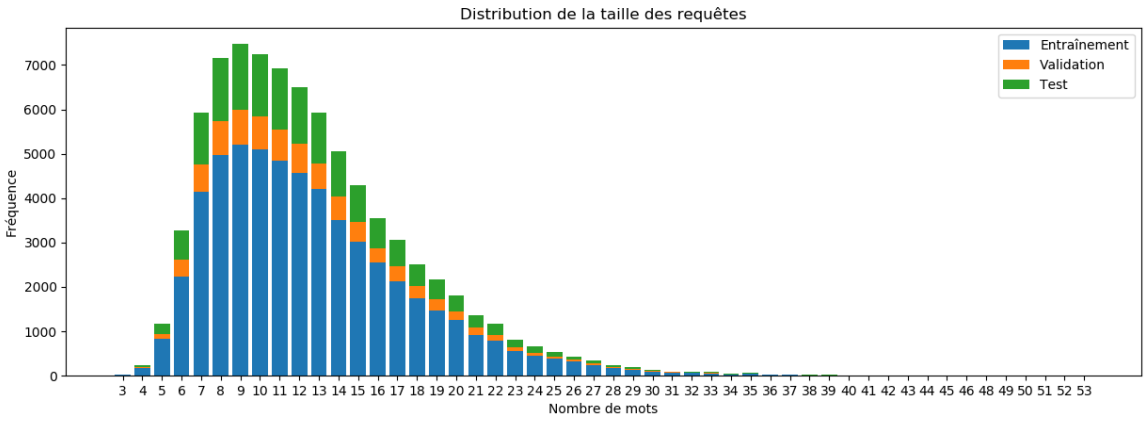


FIGURE 1 – Diagramme représentant la distribution des requêtes en français selon leur longueur

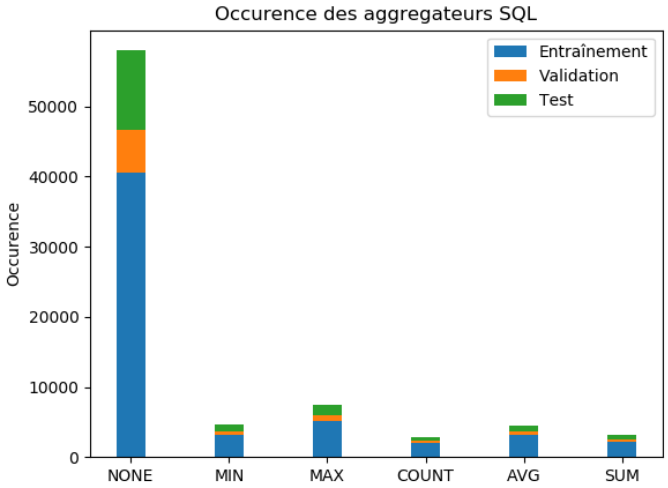


FIGURE 2 – Diagramme représentant la proportion de chaque agrégateur SQL dans les différentes partitions du jeu de données

# 3 Présentation du modèle TAPAS

Avec les progrès récents de l'apprentissage profond appliqué à la compréhension du langage naturel, des prototypes de logiciels grand public cherchant à intégrer une interrogation en langage naturel pour assurer une interaction homme-machine plus naturelle ont vu le jour. Les services proposés restent cependant extrêmement limités car ils ne prennent en compte que des structures de données spécifiques, et les interactions restent encore très éloignées d'une interaction en langage naturel. Dans le même temps, de nouvelles architectures neuronales permettent de franchir des étapes importantes pour déterminer les réponses à donner à des questions exprimées en langue naturelle. En particulier, les architectures à base de transformers telles que le BERT de (Devlin *et al.*, 2019) ont apporté des progrès notables. Pourrions-nous tirer parti de ces avancées pour interagir avec les données tabulaires ? Récemment, Google Research (Herzig *et al.*, 2020) a dévoilé TAPAS (Table parser), un modèle basé sur l'architecture BERT qui traite les questions et réponses pour des ensembles de données tabulaires.

Au lieu de créer un modèle contraint à une structure de table spécifique, Google a fait le choix d'une approche plus globale en créant un réseau de neurones adapté à toute forme de jeu de données tabulaires. Son modèle TAPAS réutilise l'architecture de l'encodeur BERT, en y ajoutant des plongements supplémentaires. L'ajout le plus notable au modèle de base BERT est l'intégration d'informations supplé-mentaires pour l'encodage de l'entrée textuelle. Tapas exploite les incorporations apprises pour les index de ligne et de colonne ainsi que pour un index de rang spécial qui représente l'ordre des éléments dans les colonnes numériques. L'architecture obtenue surpasse actuellement les autres modèles pour l'interrogation en langage naturel de données tabulaires.

Le modèle TAPAS est assez similaire à BERT mais il en diffère par l'ajout à son *tokenizer* des plongements des positions relatives, ainsi que sept tokens modélisant les tables. TAPAS est pré-appris sur une tâche de modèle masqué [1], à l'aide de millions de tables venant de la version anglaise de Wikipedia et les textes correspondants.

Enfin, TAPAS est entraîné plus finement sur une tâche des réponses aux questions. Le modèle cherche alors à prédire deux choses : les cellules correctes associées à la réponse et l'agrégateur correspondant.

# 4 Application et résultats

## 4.1 Apprentissage du modèle

Outre les ajouts et modifications apportés au modèle BERT originel décrit précédemment, TAPAS suit un protocole d'apprentissage en trois étapes qui s'appuie sur plusieurs jeux de données. La première étape consiste en un pré-entraînement sur un jeu de données de 6,2 millions de tables anglophones extraites de Wikipedia suivant le modèle de masquage proposé dans BERT (Devlin *et al.*, 2019). Le but ici est d'initialiser l'entraînement du modèle à partir du contexte constitué des éléments qui composent la table traitée : la tâche consiste à retrouver certains éléments masqués de ce contexte.

Cette étape de pré-apprentissage est suivie d'une étape d'ajustement fin (fine tuning) qui finalise

1. Masked language modeling (MLM)

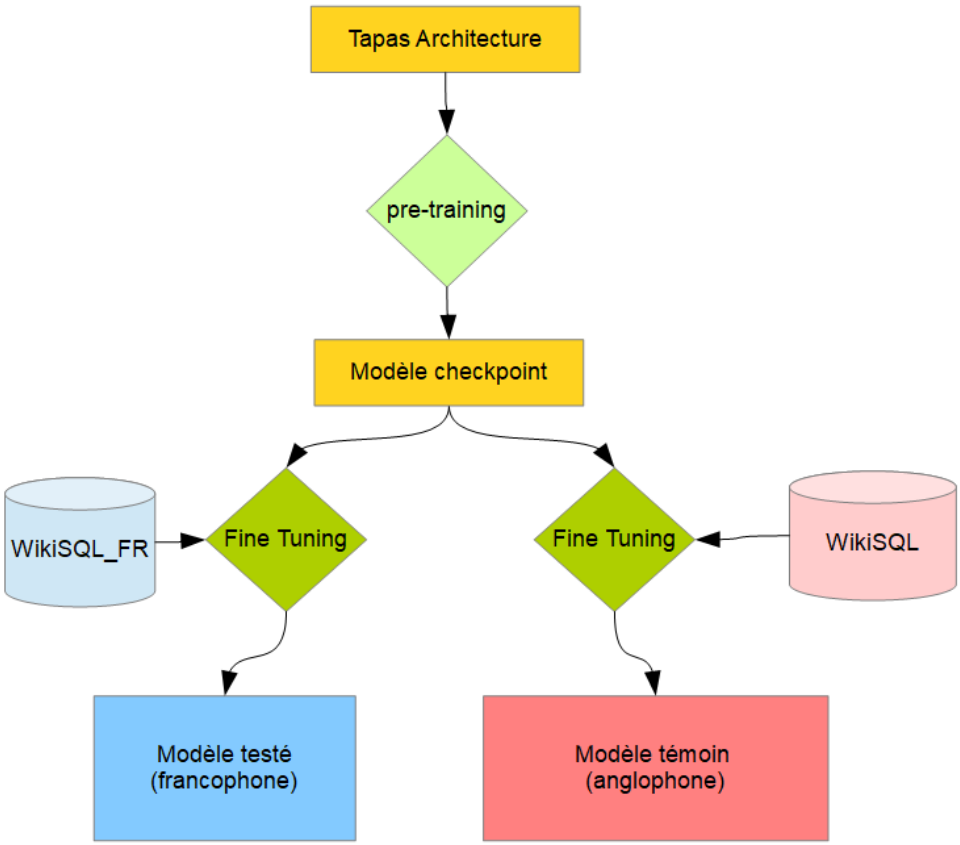


FIGURE 3 – Schéma représentant le protocole d'obtention des modèles

l'apprentissage du modèle sur une tâche spécifique. Les données utilisées pour cet entraînement sont alors similaires au jeu de tests pour évaluer le modèle final obtenu.

Dans leurs travaux les plus récents sur le modèle (Eisenschlos *et al.*, 2020), les auteurs ont ajouté une étape intermédiaire d'apprentissage. Pour notre expérimentation, nous sommes partis de ce modèle ayant bénéficié de ce pré-entraînement supplémentaire.

## 4.2 Expérimentation sur les données francophones

Le but de l'expérimentation est de déterminer dans quelle mesure ce nouveau jeu de données francophones impacte les performances du modèle TAPAS. Ainsi l'un des modèles TAPAS entraîné sur WikiSQL pour la langue anglaise sert de modèle témoin. À cela on compare un modèle suivant la même architecture et ayant reçu le même pré-entraînement, mais dont l'entraînement fin a été réalisé sur le nouveau jeu de données francophones. Les deux jeux de données sont alors testés sur leurs jeux de données respectifs. Un schéma récapitule ce processus en figure 3, les indicateurs de sont les mêmes que ceux décrits dans l'article originel de Tapas.

Ainsi afin d'avoir les conditions expérimentales les plus similaires entre ces deux modèles, nous avons pris comme point de contrôle (checkpoint) d'apprentissage le modèle *base* de TAPAS que nous avons entraîné sur les jeux de données respectifs sur 100000 et 200000 pas de batch 4 pour un total de 400000 et 800000 mises à jour. On obtient alors les résultats en tableau 3. Les méthodes de test

sont les mêmes que celles utilisées par TAPAS, et les modèles sont testés sur les mêmes données que celles utilisées pour l'apprentissage fin.

| | Modèle anglophone | Modèle francophone |
|---|---|---|
| 100000 pas dev/test ex(acc) | 0.8248/0.7993 | 0.5366 / 0.5195 |
| 200000 pas dev/test ex(acc) | 0.8246 / 0.7979 | 0.6028 / 0.5864 |
| résultats finaux [2]dev/test ex(acc) | 0.8859 | 0.5967/0.5774 |

TABLE 3 – Exactitude (accuracy) sur le jeu de tests et développement des deux modèles anglophone et francophone.

On peut alors voir plusieurs différences entre les résultats des deux modèles. Tout d'abord le modèle francophone semble avoir, après 200000 pas, des résultats 20% inférieurs à ceux du modèle anglophone. De plus le modèle de langue française semble bien plus sensible à cette dernière phase d'apprentissage que le modèle anglophone. Cette différence dans la vitesse de convergence dans la deuxième phase peut s'expliquer par le changement de langue entre la phase de pré-apprentissage et la phase d'apprentissage fin, les résultats finaux présentés pour notre modèle francophone ayant atteint une valeur asymptotique.

# 5 Conclusion et travaux futurs

Si les résultats propres de l'expérience ne permettent pas d'en déduire pour l'instant une réelle efficacité du modèle en situation réelle, ils permettent néanmoins d'encourager cette approche de traduction de jeu de données. En effet on peut espérer de meilleurs résultats avec un apprentissage plus long, ou un jeu de données traduit et relu, ce qui allégerait la tâche d'expertise humaine nécessaire à l'obtention d'un jeu de données pour cette tâche de traitement de requêtes en français sur des données tabulaires.

De plus ces résultats nous éclairent sur l'usage de l'apprentissage fin, qui permet d'une part d'adapter plus facilement de larges modèles à de nouvelles langues, et d'autre part d'éviter de travailler sur des modèles complets souvent très volumineux ; dans notre cas les données de pré-apprentissage forment un corpus de 6.2 millions de tables.

Enfin, des résultats complémentaires permettront d'affiner les conclusions et perspectives de cette étude, comme sur l'influence de la traduction sur les performances d'exactitude. On peut aussi imaginer un modèle suivant la même architecture mais prenant en compte des données francophones sur chaque étape de l'apprentissage.

# Références

AIKEN M. (2019). An updated evaluation of google translate accuracy. *Studies in Linguistics and Literature*, **3**, p253. DOI : 10.22158/sll.v3n3p253.

2. Tels que présentés dans l'article de (Eisenschlos *et al.*, 2020), après entraînement sur TPU.

BAIK C., JAGADISH H. V. & LI Y. (2019). Bridging the semantic gap with sql query logs in natural language interfaces to databases. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, p. 374–385.

DEVLIN J., CHANG M.-W., LEE K. & TOUTANOVA K. (2019). Bert : Pre-training of deep bidirectional transformers for language understanding.

EISENSCHLOS J. M., KRICHENE S. & MÜLLER T. (2020). Understanding tables with intermediate pre-training.

HERZIG J., NOWAK P. K., MÜLLER T., PICCINNO F. & EISENSCHLOS J. (2020). TaPas : Weakly supervised table parsing via pre-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, p. 4320–4333, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.acl-main.398.

IYYER M., TAU YIH W. & CHANG M.-W. (2017). Search-based neural structured learning for sequential question answering. In *ACL (1)*, p. 1821–1831.

KABBADJ A. (2018). Something new in french text mining and information extraction (universal chatbot) : Largest qa french training dataset (110 000+). [Online ; posted 11-November-2018].

LYU Q., CHAKRABARTI K., HATHI S., KUNDU S., ZHANG J. & CHEN Z. (2020). Hybrid ranking network for text-to-sql.

PASUPAT P. & LIANG P. (2015). Compositional semantic parsing on semi-structured tables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, p. 1470–1480, Beijing, China : Association for Computational Linguistics. DOI : 10.3115/v1/P15-1142.

RAJPURKAR P., JIA R. & LIANG P. (2018). Know what you don't know : Unanswerable questions for squad.

WANG A., SINGH A., MICHAEL J., HILL F., LEVY O. & BOWMAN S. R. (2019). Glue : A multi-task benchmark and analysis platform for natural language understanding.

ZHONG V., XIONG C. & SOCHER R. (2017). Seq2sql : Generating structured queries from natural language using reinforcement learning. *CoRR*, **abs/1709.00103**.